



Τεχνικές Προγραμματισμού και Χρήση Λογισμικού Η/Υ στις Κατασκευές

Ενότητα 3: Διαδικασίες λογικών αποφάσεων και βρόγχων εργασιών

Αναστάσιος Σέξτος
Αναπληρωτής Καθηγητής Α.Π.Θ.
Τμήμα Πολιτικών Μηχανικών



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



ΕΣΠΑ
2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ

Άδειες Χρήσης

- Το παρόν εκπαιδευτικό υλικό υπόκειται σε άδειες χρήσης Creative Commons.
- Για εκπαιδευτικό υλικό, όπως εικόνες, που υπόκειται σε άλλου τύπου άδειας χρήσης, η άδεια χρήσης αναφέρεται ρητώς.



Χρηματοδότηση

- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.
- Το έργο «Ανοικτά Ακαδημαϊκά Μαθήματα στο Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού.
- Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.





ΑΡΙΣΤΟΤΕΛΕΙΟ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΘΕΣΣΑΛΟΝΙΚΗΣ

ΑΝΟΙΚΤΑ
ΑΚΑΔΗΜΑΙΚΑ
ΜΑΘΗΜΑΤΑ



Διαδικασίες λογικών αποφάσεων και βρόγχων εργασιών

Περιεχόμενα ενότητας

1. Προγραμματισμός υπό συνθήκες
2. Διαδικασίες λογικών αποφάσεων και βρόγχων εργασιών
3. Εργασία με βρόγχους
4. Εφαρμογή



Σκοποί ενότητας

- Να παρουσιαστούν οι βασικές αρχές των εργασιών με βρόγχους
- Να πραγματοποιηθεί μία πλήρης απλή εφαρμογή



Προγραμματισμός υπό συνθήκες

Η αξία του να έχει τη δυνατότητα ο χρήστης να προγραμματίσει μια σειρά εργασιών έγκειται σε δύο βασικές δυνατότητες:

- α) στο να διαμορφώνει **«έξυπνη» ροή διαδικασιών** θέτοντας λογικές συνθήκες που ελέγχουν αν θα ακολουθηθεί στο επόμενο βήμα η μία ή η άλλη διεργασία
- β) στο να πραγματοποιούνται **κυκλικές (επαναλαμβανόμενες) διαδικασίες** με σύντομες εντολές οι οποίες διατυπώνονται μία φορά μόνο.

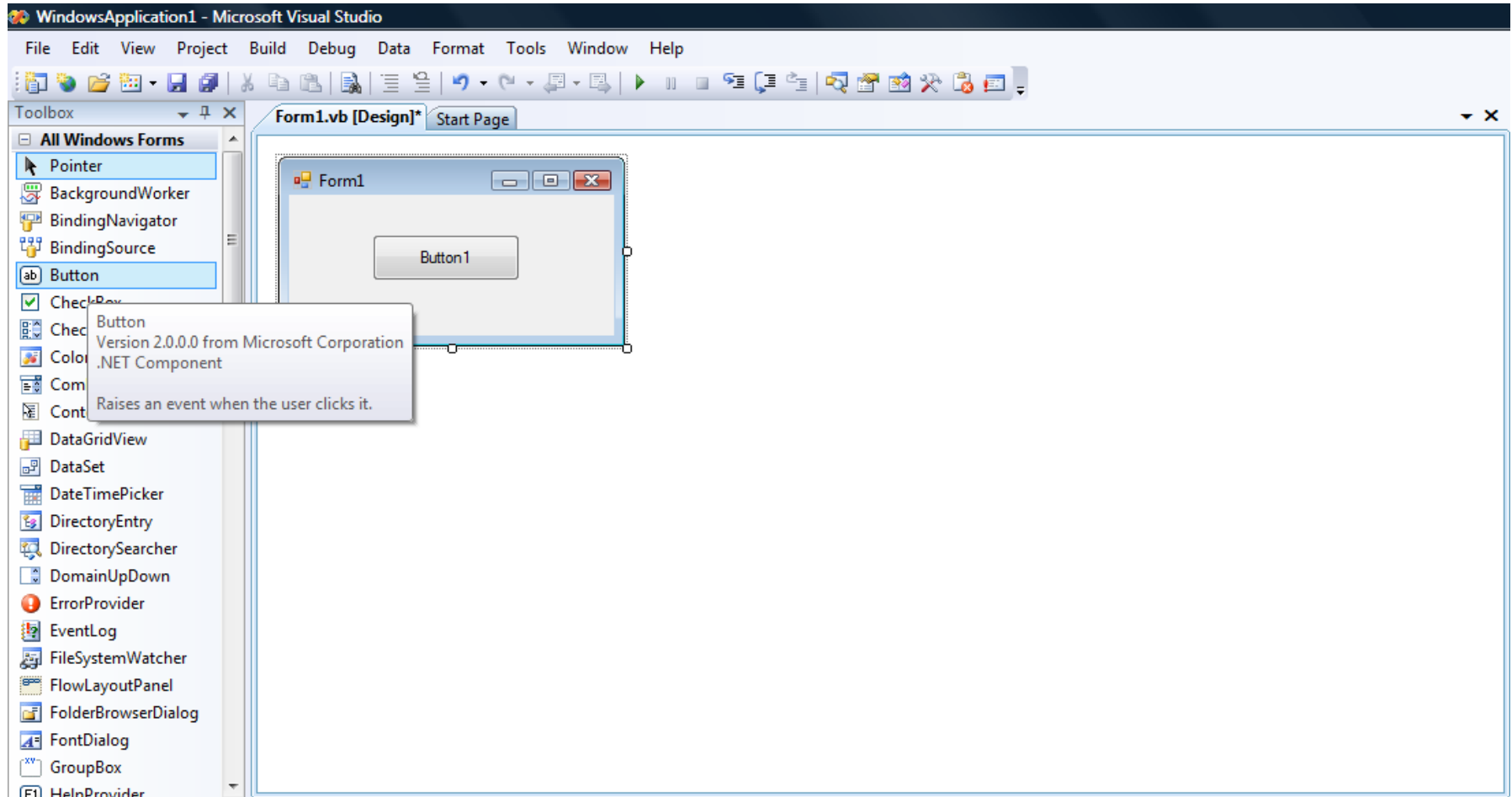


Προγραμματισμός υπό συνθήκες

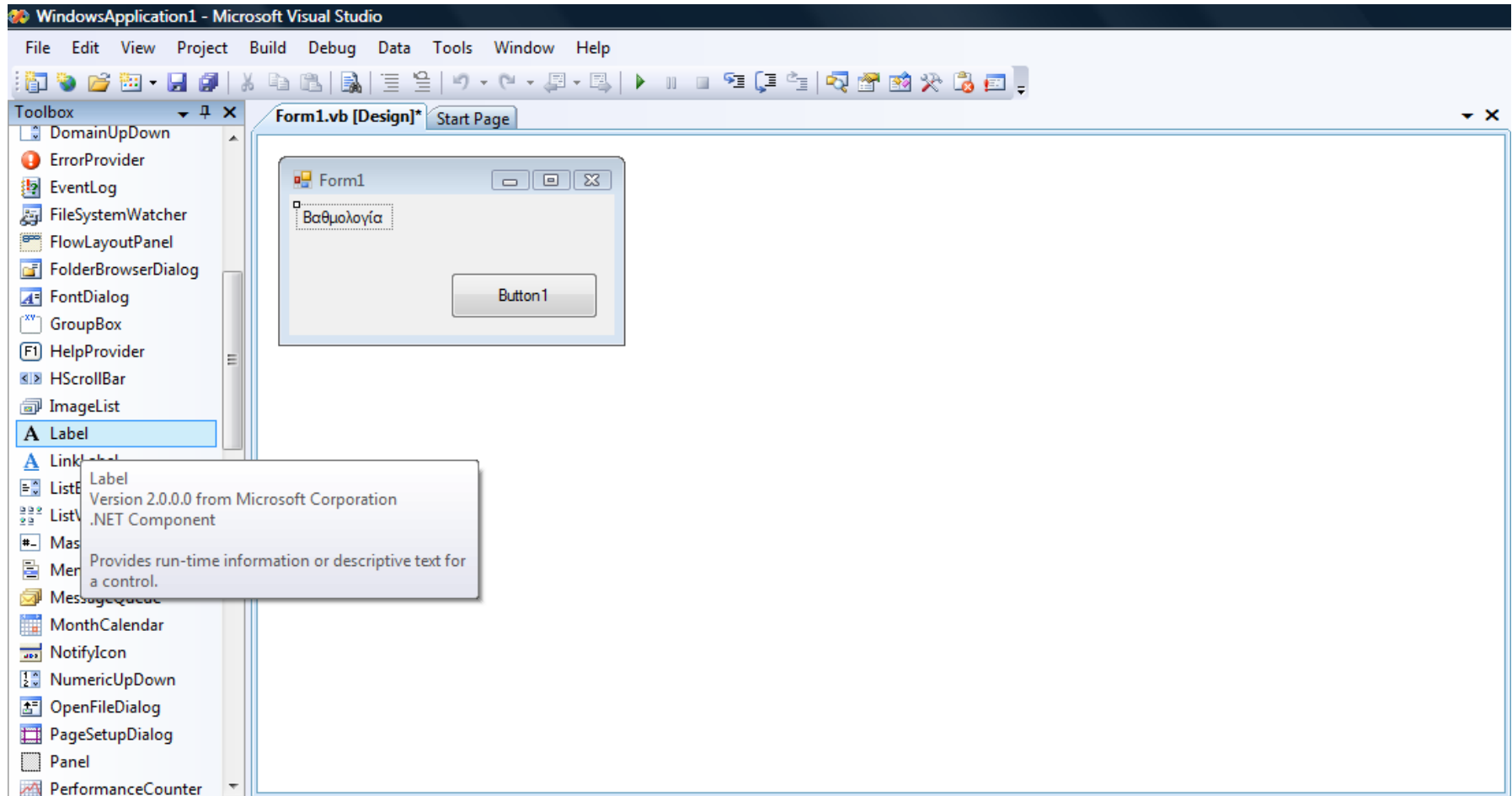
Οι δυνατότητες αυτές καθιστούν τον προγραμματισμό προγραμμάτων και εφαρμογών πολύτιμο και αναντικατάστατο από γενικά εμπορικά προγράμματα, ακόμη και από προγράμματα λογιστικών φύλλων των οποίων η κύρια εφαρμογή είναι η διαχείριση δεδομένων.



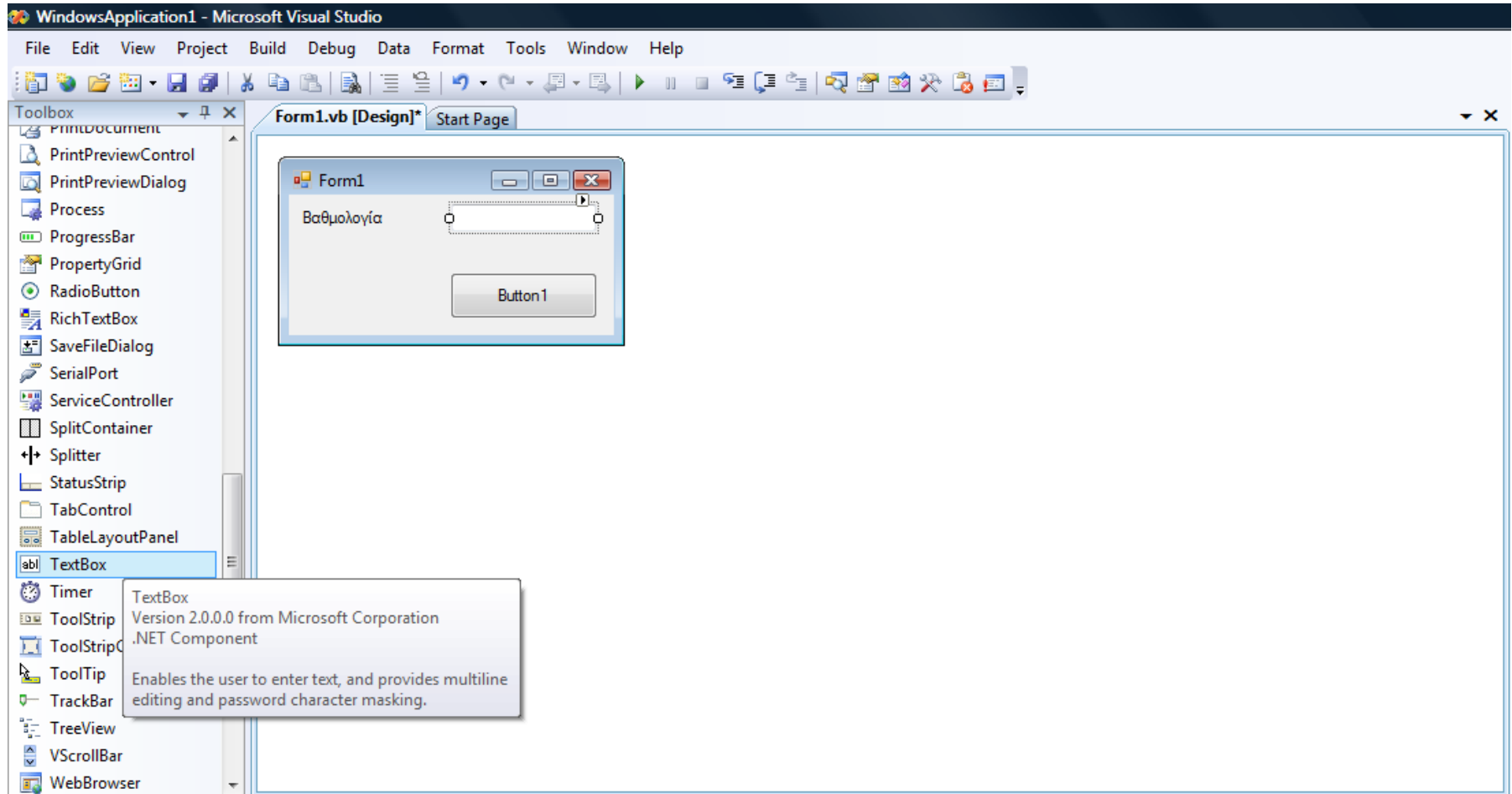
Προγραμματισμός υπό συνθήκες



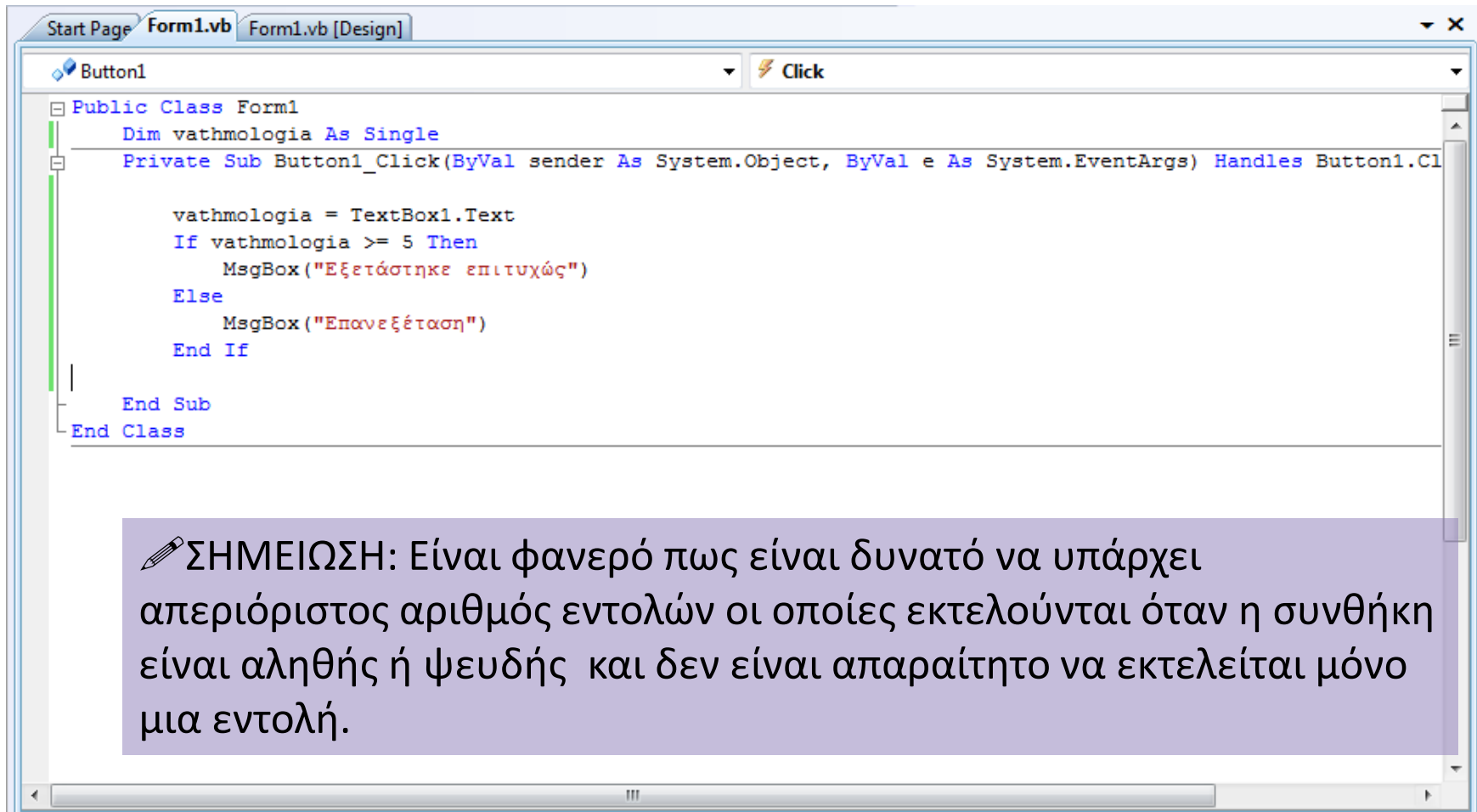
Προγραμματισμός υπό συνθήκες



Προγραμματισμός υπό συνθήκες




Προγραμματισμός υπό συνθήκες



```
Public Class Form1
    Dim vathmologia As Single
    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click

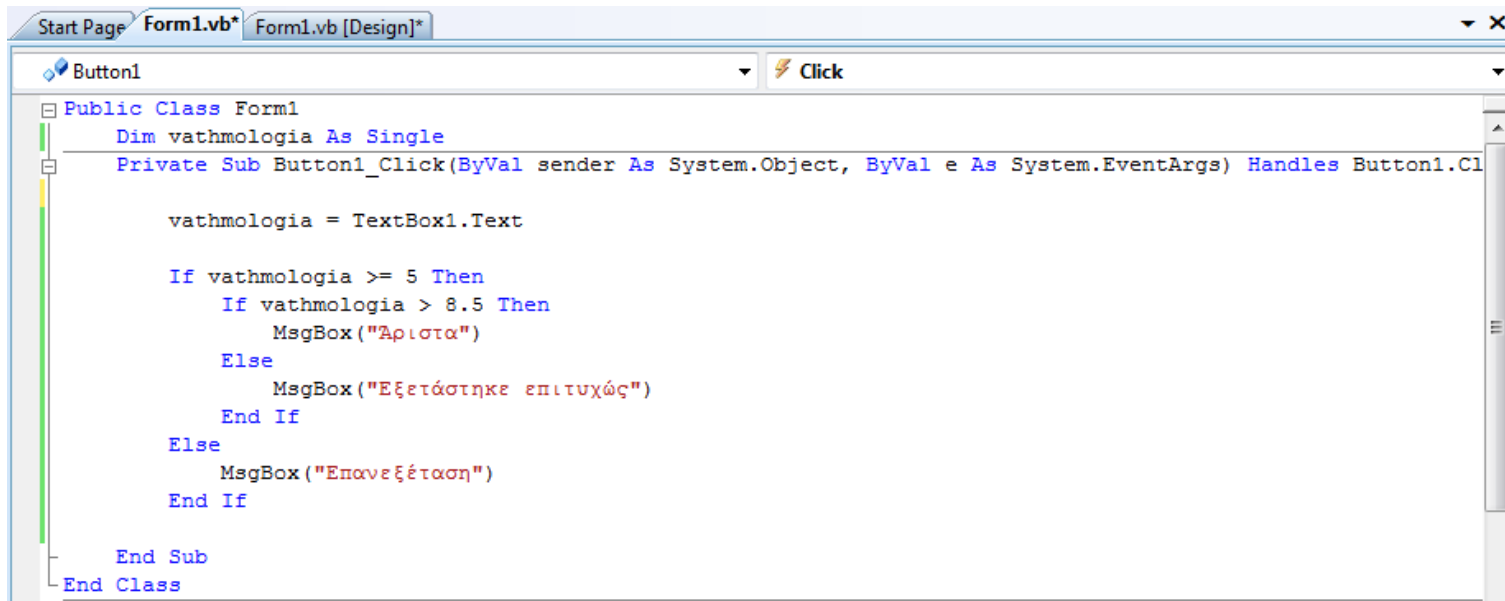
        vathmologia = TextBox1.Text
        If vathmologia >= 5 Then
            MsgBox("Εξετάστηκε επιτυχώς")
        Else
            MsgBox("Επανεξέταση")
        End If
    End Sub
End Class
```

 ΣΗΜΕΙΩΣΗ: Είναι φανερό πως είναι δυνατό να υπάρχει απεριόριστος αριθμός εντολών οι οποίες εκτελούνται όταν η συνθήκη είναι αληθής ή ψευδής και δεν είναι απαραίτητο να εκτελείται μόνο μια εντολή.

Προγραμματισμός υπό συνθήκες

Χρήση εντολής if εντός άλλης εντολής if

Στις περισσότερες πρακτικές περιπτώσεις όταν ικανοποιείται (ή αντίστοιχα δεν πληρείται) μια συνθήκη, υπάρχει δεύτερος λογικός έλεγχος μιας δεύτερης συνθήκης η οποία επίσης διατυπώνεται μέσω της εντολής if.



```
Public Class Form1
    Dim vathmologia As Single
    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click

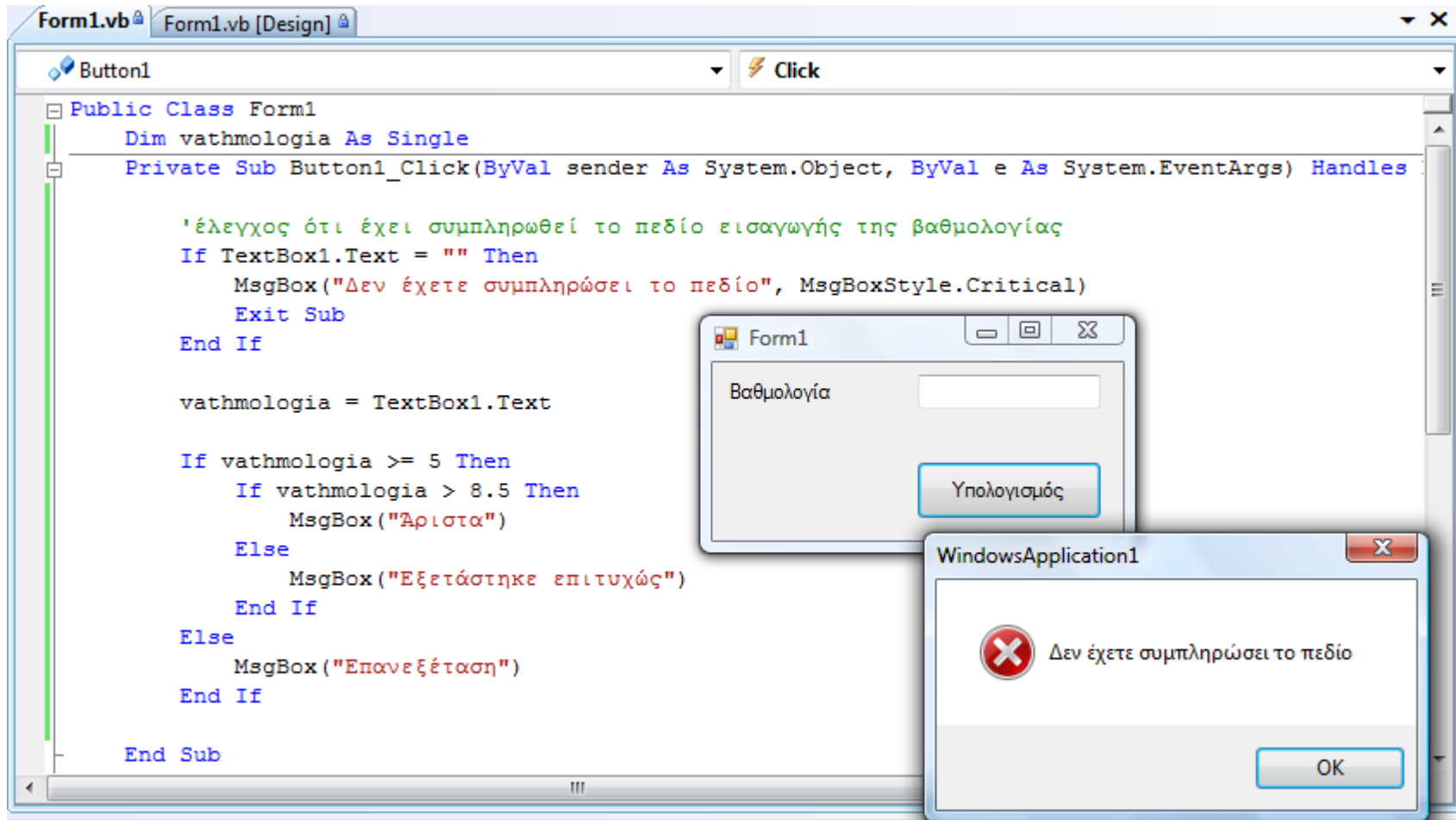
        vathmologia = TextBox1.Text

        If vathmologia >= 5 Then
            If vathmologia > 8.5 Then
                MsgBox("Άριστα")
            Else
                MsgBox("Εξετάστηκε επιτυχώς")
            End If
        Else
            MsgBox("Επανεξέταση")
        End If

    End Sub
End Class
```



Προγραμματισμός υπό συνθήκες



The screenshot displays the Visual Studio IDE with a VB.NET project named 'Form1'. The code in the 'Form1.vb' file implements a grade calculation logic with conditional checks. The code is as follows:

```
Public Class Form1
    Dim vathmologia As Single
    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
        'έλεγχος ότι έχει συμπληρωθεί το πεδίο εισαγωγής της βαθμολογίας
        If TextBox1.Text = "" Then
            MsgBox("Δεν έχετε συμπληρώσει το πεδίο", MsgBoxStyle.Critical)
            Exit Sub
        End If

        vathmologia = TextBox1.Text

        If vathmologia >= 5 Then
            If vathmologia > 8.5 Then
                MsgBox("Άριστα")
            Else
                MsgBox("Εξετάστηκε επιτυχώς")
            End If
        Else
            MsgBox("Επανεξέταση")
        End If
    End Sub
End Class
```

Two application windows are overlaid on the code editor. The top window, titled 'Form1', shows a text box labeled 'Βαθμολογία' and a button labeled 'Υπολογισμός'. The bottom window, titled 'WindowsApplication1', is a message box with a red 'X' icon and the text 'Δεν έχετε συμπληρώσει το πεδίο', with an 'OK' button at the bottom.



Προγραμματισμός υπό συνθήκες

```
WindowsApplication1 - Microsoft Visual Studio
File Edit View Project Build Debug Data Tools Window Help
Toolbox Start Page Form1.vb* Form1.vb [Design]*
General
There are no usable controls in this group. Drag an item onto this text to add it to the toolbox.
Button1 Click
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Click
    If TextBox1.Text = "" Then ' Να θυμάμαι ότι "" σημαίνει KENO
        MsgBox("παρακαλώ συμπληρώστε τιμή στο πεδίο εισαγωγής")
    Else
        Vathmologia = TextBox1.Text
        If Vathmologia >= 5 Then
            If Vathmologia > 8.5 Then
                If Vathmologia > 10 Then
                    MsgBox("παρακαλώ διορθώστε βαθμολογία ώστε <10")
                Else
                    MsgBox("άριστα")
                End If
            Else
                MsgBox("εξέταση επιτυχής", MsgBoxStyle.Exclamation)
            End If
        Else
            MsgBox("εξέταση ανεπιτυχής - απαιτείται επανεξέταση", MsgBoxStyle.Critical)
        End If
    End If
End Sub
```



Διαδικασίες λογικών αποφάσεων και βρόγχων εργασιών

Η εντολή του άμεσου If:

Μπορούμε να χρησιμοποιήσουμε και τη *συνάρτηση του άμεσου If (Immediate If), την IIF()*, η οποία επιστρέφει μια τιμή που μπορούμε να αποθηκεύσουμε απευθείας σε μια μεταβλητή και δέχεται τρία ορίσματα : τη συνθήκη που ελέγχεται, την τιμή που θα επιστραφεί αν η συνθήκη είναι αληθής και την τιμή που θα επιστραφεί αν η συνθήκη είναι ψευδής. Η εντολή αυτή μοιάζει σε μεγάλο βαθμό με το γνωστό τρόπο που έχουμε συνηθίσει να θέτουμε λογικά ερωτήματα στα προγράμματα λογιστικών φύλλων όπως το Excel.



Διαδικασίες λογικών αποφάσεων και βρόγχων εργασιών

ΠΑΡΑΔΕΙΓΜΑ:

$$b = \text{IIF}(b \geq 10, 10, b)$$

Δηλαδή η μεταβλητή b δεν επιτρέπεται να υπερβεί την τιμή 10 εφόσον αν $b > 10$ ισούται με 10, διαφορετικά παραμένει b .



Προγραμματισμός υπό συνθήκες

Η εντολή **Select case** είναι μια ευέλικτη εντολή με την οποία είναι δυνατή η επιλογή και εκτέλεση μίας ή περισσότερων εντολών ανάλογα με την τιμή που έχει μια παράσταση ή μεταβλητή. Ασφαλώς υπάρχει και η συμπληρωματική εντολή **Case Else** η οποία εκτελείται όταν δεν ισχύει καμία από τις υπάρχουσες συνθήκες. Εξ' ορισμού η εντολή κλείνει με τη διατύπωση **End Select**.



Προγραμματισμός υπό συνθήκες

ΠΑΡΑΔΕΙΓΜΑ:

Εύρεση μέγιστης εδαφικής επιτάχυνσης ως συνάρτηση της ζώνης σεισμικής επικινδυνότητας κατά ΕΑΚ2000.

Select Case SeismicZone

Case 1: PGA = 0.16

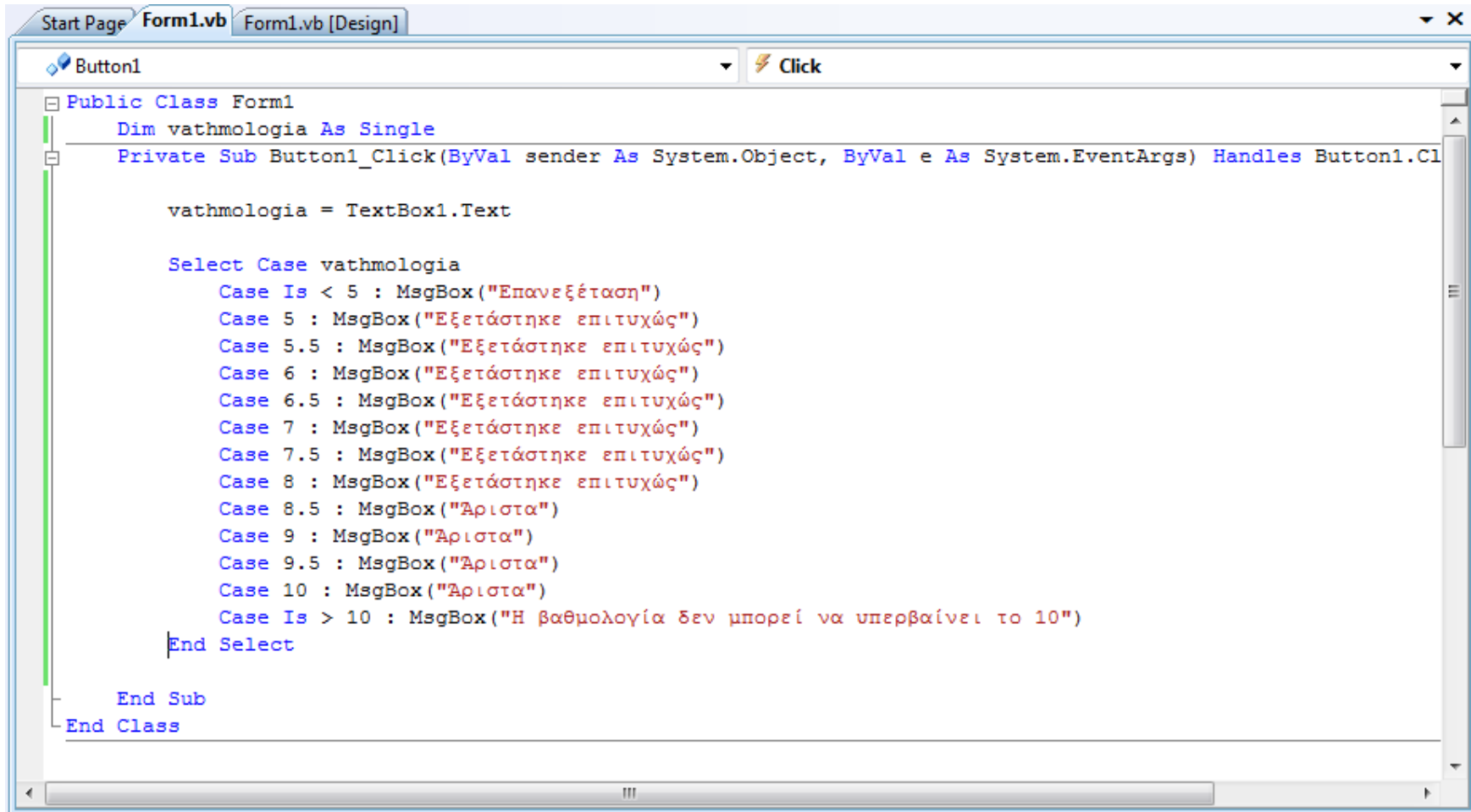
Case 2: PGA = 0.24

Case 3: PGA = 0.36

End Select



Προγραμματισμός υπό συνθήκες



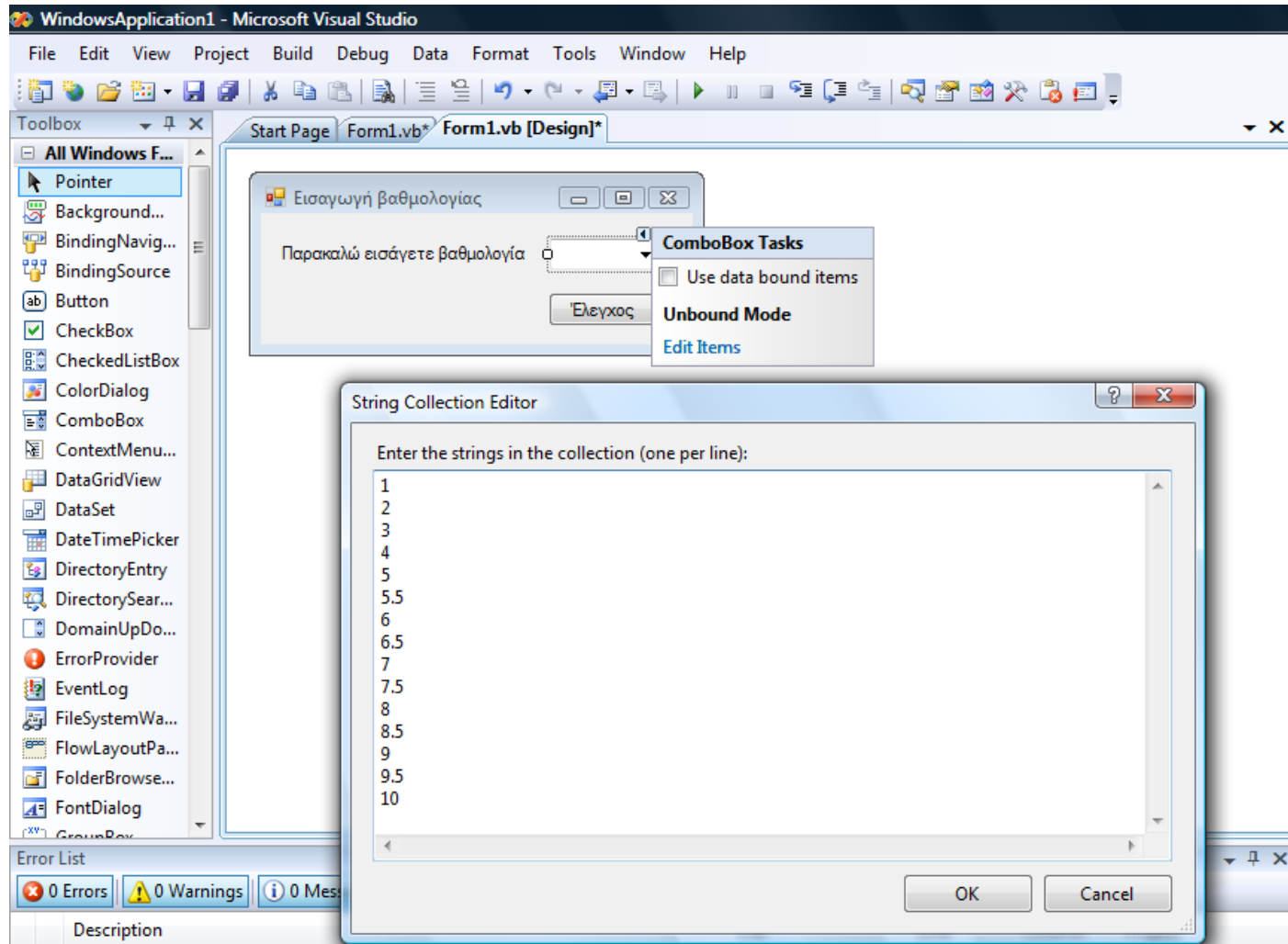
```
Public Class Form1
    Dim vathmologia As Single
    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click

        vathmologia = TextBox1.Text

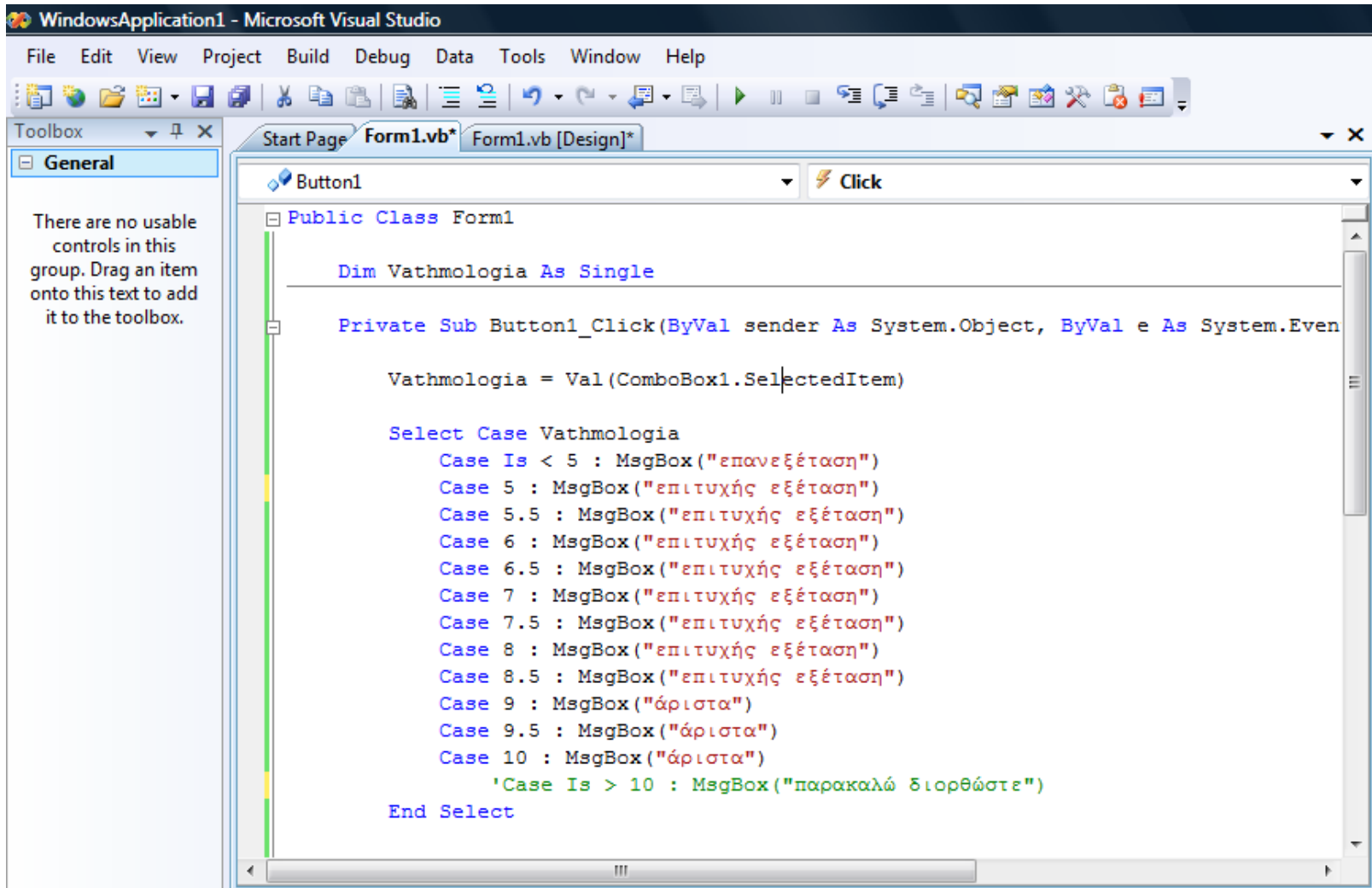
        Select Case vathmologia
            Case Is < 5 : MsgBox("Επανεξέταση")
            Case 5 : MsgBox("Εξετάστηκε επιτυχώς")
            Case 5.5 : MsgBox("Εξετάστηκε επιτυχώς")
            Case 6 : MsgBox("Εξετάστηκε επιτυχώς")
            Case 6.5 : MsgBox("Εξετάστηκε επιτυχώς")
            Case 7 : MsgBox("Εξετάστηκε επιτυχώς")
            Case 7.5 : MsgBox("Εξετάστηκε επιτυχώς")
            Case 8 : MsgBox("Εξετάστηκε επιτυχώς")
            Case 8.5 : MsgBox("Άριστα")
            Case 9 : MsgBox("Άριστα")
            Case 9.5 : MsgBox("Άριστα")
            Case 10 : MsgBox("Άριστα")
            Case Is > 10 : MsgBox("Η βαθμολογία δεν μπορεί να υπερβαίνει το 10")
        End Select

    End Sub
End Class
```

Προγραμματισμός υπό συνθήκες



Προγραμματισμός υπό συνθήκες



```
WindowsApplication1 - Microsoft Visual Studio
File Edit View Project Build Debug Data Tools Window Help
Toolbox
General
There are no usable controls in this group. Drag an item onto this text to add it to the toolbox.
Start Page Form1.vb* Form1.vb [Design]*
Button1 Click
Public Class Form1
    Dim Vathmologia As Single
    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
        Vathmologia = Val(ComboBox1.SelectedItem)
        Select Case Vathmologia
            Case Is < 5 : MsgBox("επανεξέταση")
            Case 5 : MsgBox("επιτυχής εξέταση")
            Case 5.5 : MsgBox("επιτυχής εξέταση")
            Case 6 : MsgBox("επιτυχής εξέταση")
            Case 6.5 : MsgBox("επιτυχής εξέταση")
            Case 7 : MsgBox("επιτυχής εξέταση")
            Case 7.5 : MsgBox("επιτυχής εξέταση")
            Case 8 : MsgBox("επιτυχής εξέταση")
            Case 8.5 : MsgBox("επιτυχής εξέταση")
            Case 9 : MsgBox("άριστα")
            Case 9.5 : MsgBox("άριστα")
            Case 10 : MsgBox("άριστα")
            'Case Is > 10 : MsgBox("παρακαλώ διορθώστε")
        End Select
    End Sub
End Class
```

Εργασία με βρόγχους

- Ένας **βρόγχος** είναι μια συνολική διαδικασία δια της οποίας επαναλαμβάνεται μια σειρά επιμέρους δράσεων μέχρις ότου μια συνθήκη ικανοποιηθεί.
- Η δυνατότητα αυτή που παρέχουν οι βρόγχοι επαναληπτικής εκτέλεσης εργασιών αποτελεί ουσιαστικά το κυριότερο προγραμματιστικό εργαλείο για τη διαχείριση των δεδομένων, ιδιαίτερα αυτών που βρίσκονται υπό μορφή πίνακα.
- Οι βρόγχοι συνεπώς είναι στενά, αν και όχι αποκλειστικά, συνδεδεμένοι με τα διανύσματα και τους πίνακες.
- Η Visual Basic διαθέτει τις εξής τρεις βασικές δομές βρόχων: **For ... Next**, **Do ... Loop** και **While ... Wend** οι οποίες και περιγράφονται στις ενότητες που ακολουθούν.



Εργασία με βρόγχους

Ο βρόγχος For-Next

Με τη δομή **For ... Next** μπορούμε να επαναλάβουμε μια ομάδα εντολών συγκεκριμένες φορές, όπως ακριβώς γίνεται και σε άλλες γνωστές γλώσσες προγραμματισμού. Ο **μετρητής** είναι και αυτός μια μεταβλητή που λαμβάνει όλες τις τιμές από την αρχική μέχρι την τελική τιμή ανά ένα καθορισμένο βήμα (step), επιστρέφοντας στην αρχή κάθε φορά που συναντά το τέλος του βρόχου (το οποίο δηλώνεται με την εντολή *Next*).



Εργασία με βρόγχους

ΠΑΡΑΔΕΙΓΜΑ:

```
For i = 1 To 10      'To Step=1 εξ ορισμού  
    MsgBox 'Αυτή είναι η επανάληψη με αριθμό' & i  
Next i
```

Αντίστοιχα θα μπορούσε η εκτέλεση να ακολουθεί αντίστροφη μέτρηση δηλαδή:

```
For i = 10 To 1 Step -1      'To Step= - 1  
    MsgBox 'Αυτή είναι η επανάληψη με αριθμό' & i  
Next i
```



Εργασία με βρόγχους

Ο βρόγχος Do-Loop

Μια πρόταση **Do ... Loop** είναι ένας βρόγχος υπό όρους. Με άλλα λόγια, μια σειρά ενεργειών η οποία επαναλαμβάνεται έως ότου ικανοποιηθεί μια συγκεκριμένη συνθήκη. Οι δύο βασικοί τύποι βρόγχων είναι **Do ... While** (πραγματοποίησε ενόσω) **Do ... Until** (πραγματοποίησε έως ότου).



Εργασία με βρόγχους

ΠΑΡΑΔΕΙΓΜΑ:

Η μεταβλητή `number` αυξάνει διαδοχικά έως ότου παραμένει μικρότερη της τιμής 10:

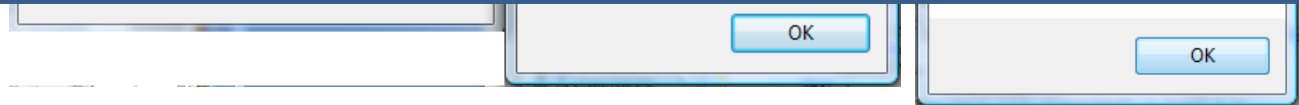
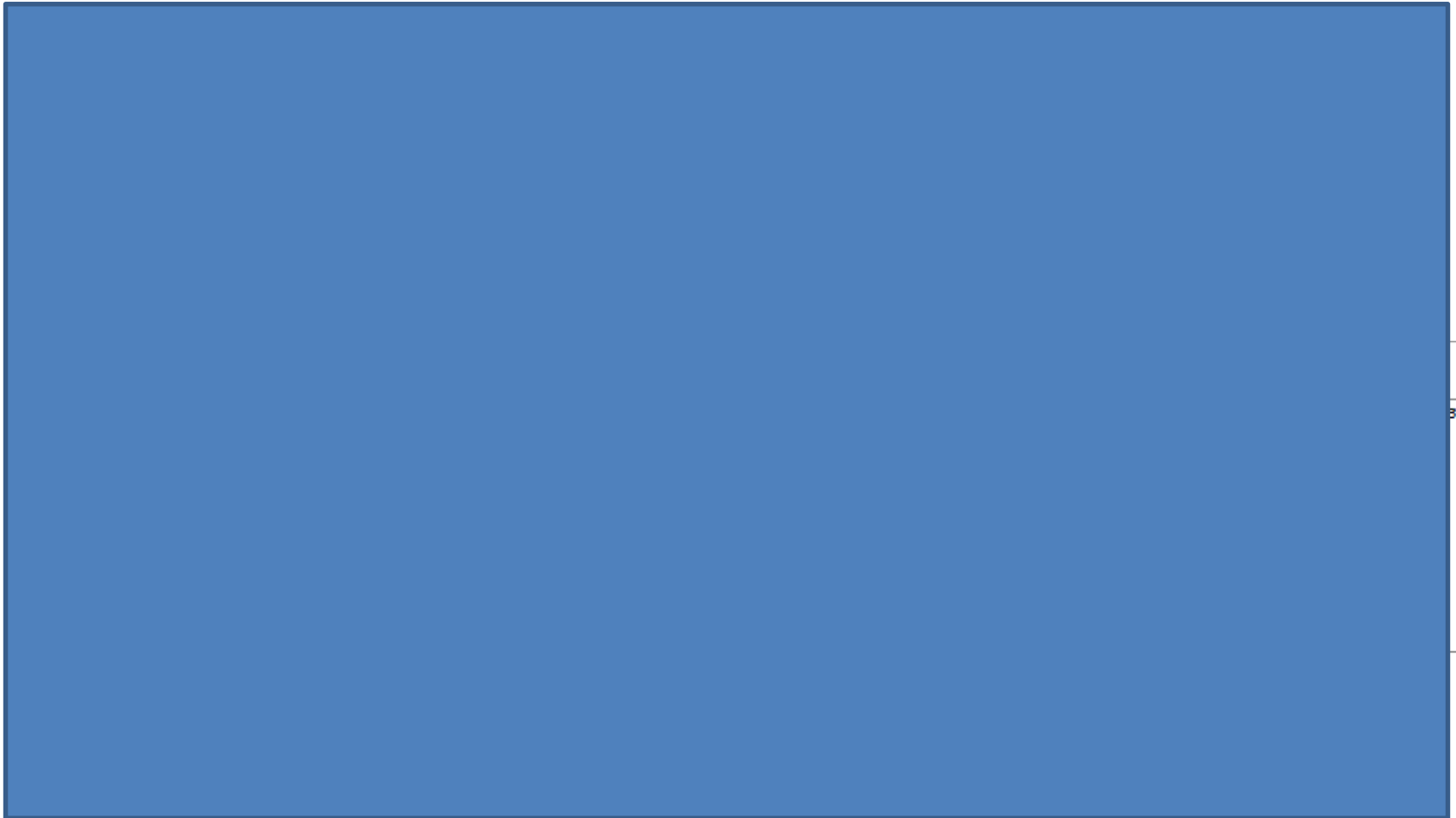
Do While `Number < 10`

`Number = Number+1`

Loop

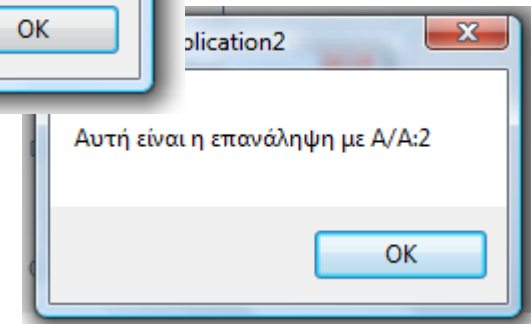
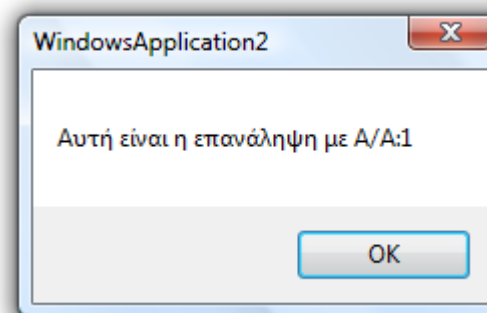
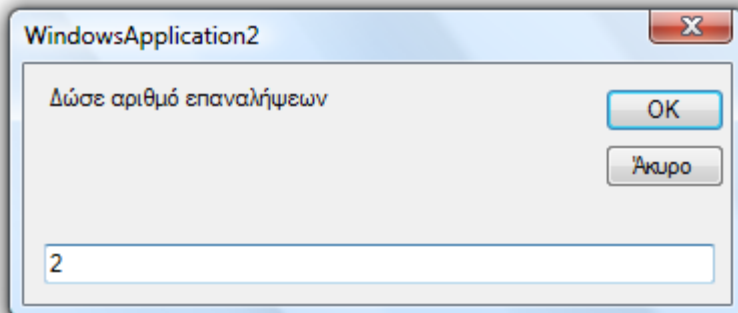


Εργασία με βρόγχους



Εργασία με βρόγχους

```
Form1 (Declarations)  
Public Class Form1  
    Dim Metritis As Integer  
    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)  
        Metritis = InputBox("Δώσε αριθμό επαναλήψεων")  
        For i = 1 To Metritis  
            MsgBox("Αυτή είναι η επανάληψη με A/A:" & i)  
        Next i  
    End Sub  
End Class
```

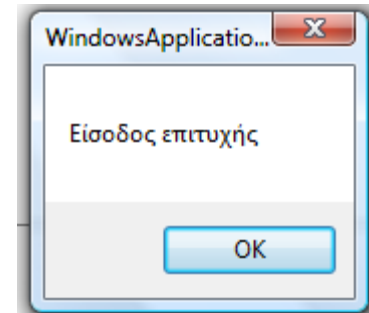
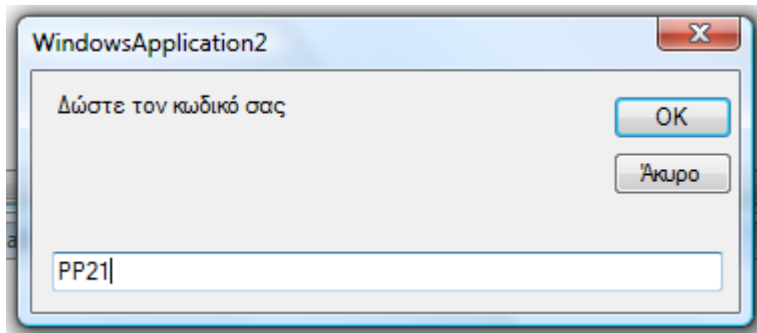


Εργασία με βρόγχους

```
Form1.vb | Form1.vb [Design]
Button1 Click
Public Class Form1
    Dim Password, Code As String

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
        Password = "PP21"
        Do
            Code = InputBox("Δώστε τον κωδικό σας")
            If Code = Password Then
                MsgBox("Είσοδος επιτυχής")
            Else
                MsgBox("Λάθος κωδικός. Παρακαλώ επαναλάβετε")
            End If
        Loop Until Code = Password

    End Sub
End Class
```



Εφαρμογή

Πρόγραμμα διαστασιολόγησης ορθογωνικής δοκού από Ο/Σ
υπό μοναξονική προέχουσα κάμψη

Υπολογισμός απαιτούμενου οπλισμού κάμψης δοκού

Ροπή κάμψης M_{sd} (kNm)
200

Αξονικό φορτίο N_{sd} (kN)
0

Ποιότητα χάλυβα
S400

Ποιότητα σκυροδέματος
C16/20

Πλάτος b (cm)
25

Ύψος h (cm)
60

Επικάλυψη (cm)
5

Διάμετρος ράβδων
14

Απαιτούμενος οπλισμός

$\mu = 0,24793$
 $\omega_1 = 0,2994$
 $\omega_2 = 0$
 $d_2/d = 0,09$
 $\rho_{min} = 0,27\%$
 $\rho_{max} = 4\%$

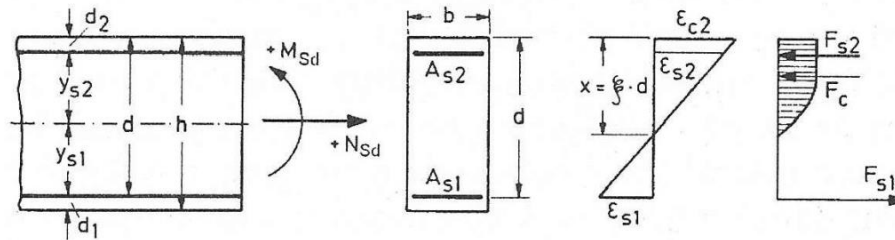
Ελάχιστος οπλισμός ανοίγματος $A_{s1,min} = 3,76 \text{ cm}^2$
Μέγιστος οπλισμός ανοίγματος $A_{s1,max} = 55 \text{ cm}^2$
Εφελκόμενος οπλισμός $A_{s1} = 12,62 \text{ cm}^2$
 $9\Phi 14 = (13,86 \text{ cm}^2)$

Υπολογισμός

Έξοδος



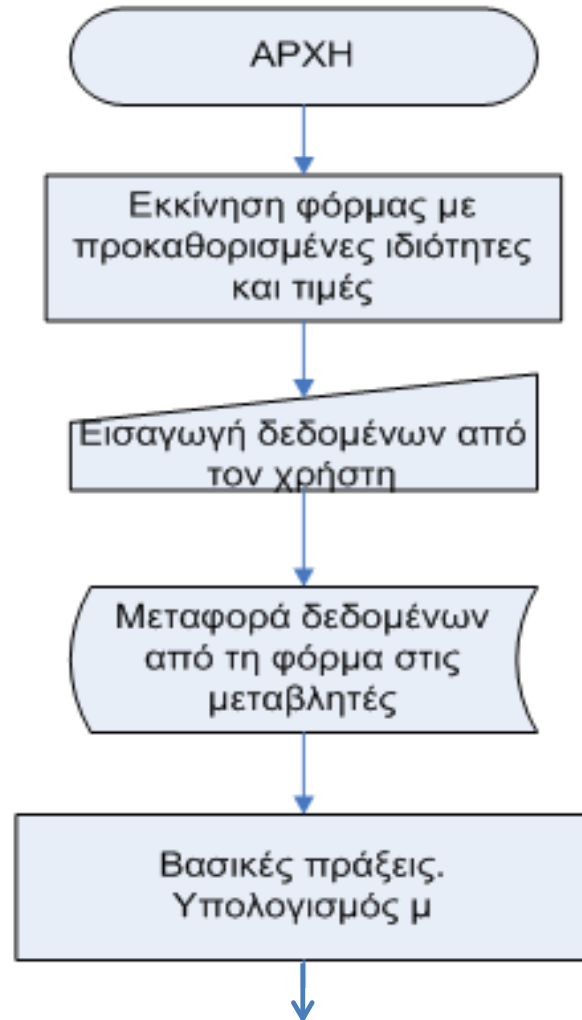
Εφαρμογή



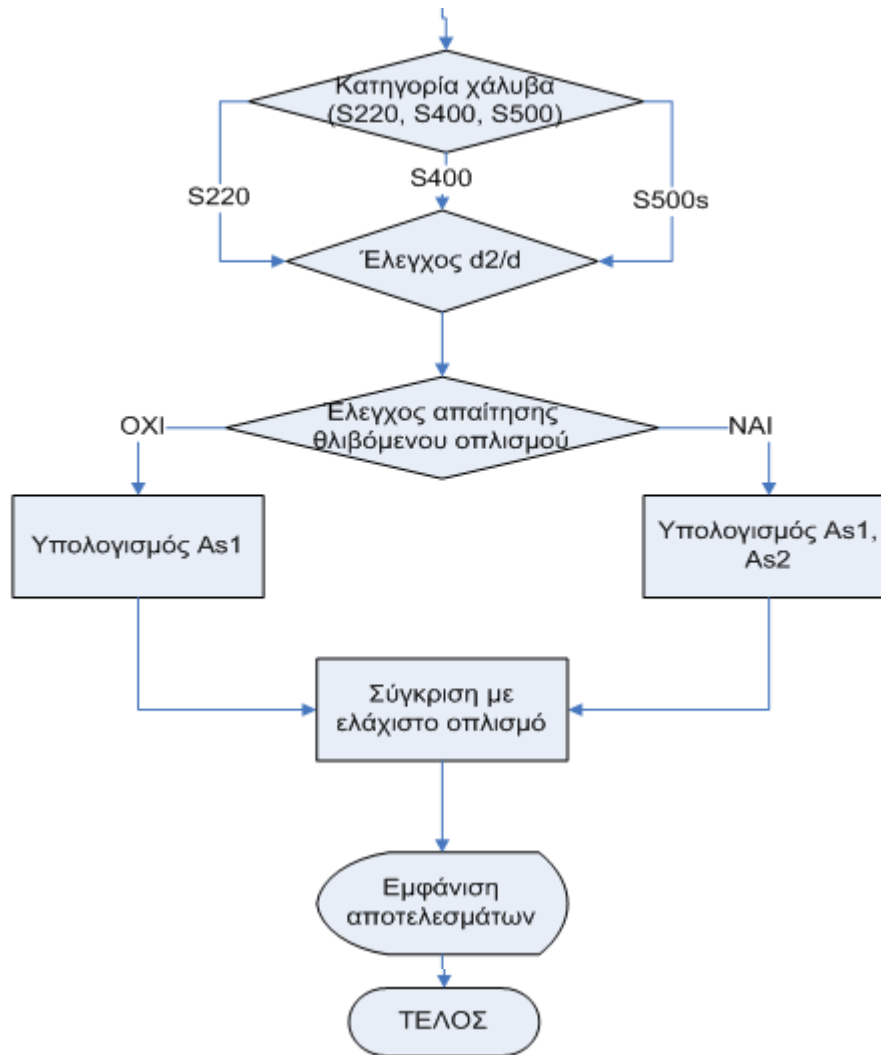
μ_{Sds}	$d_2/d = 0.05$		$d_2/d = 0.10$		$d_2/d = 0.15$		$d_2/d = 0.20$	
	ω_1	ω_2	ω_1	ω_2	ω_1	ω_2	ω_1	ω_2
0.32	0.429	0.005	0.429	0.005	0.430	0.005	0.430	0.006
0.33	0.440	0.015	0.440	0.016	0.441	0.017	0.443	0.018
0.34	0.450	0.026	0.451	0.027	0.453	0.029	0.455	0.031
0.35	0.461	0.036	0.463	0.038	0.465	0.041	0.468	0.043
0.36	0.471	0.047	0.474	0.049	0.477	0.052	0.480	0.056
0.37	0.482	0.057	0.485	0.061	0.489	0.064	0.493	0.068
0.38	0.492	0.068	0.496	0.072	0.500	0.076	0.505	0.081
0.39	0.503	0.078	0.507	0.083	0.512	0.088	0.518	0.093
0.40	0.513	0.089	0.518	0.094	0.524	0.099	0.530	0.106
0.41	0.524	0.099	0.529	0.105	0.536	0.111	0.543	0.118
0.42	0.534	0.110	0.541	0.116	0.547	0.123	0.555	0.131
0.43	0.545	0.120	0.552	0.127	0.559	0.135	0.568	0.143
0.44	0.555	0.131	0.563	0.138	0.571	0.146	0.580	0.156
0.45	0.566	0.142	0.574	0.149	0.583	0.158	0.593	0.168
0.46	0.577	0.152	0.585	0.161	0.594	0.170	0.605	0.181
0.47	0.587	0.163	0.596	0.172	0.606	0.182	0.618	0.193
0.48	0.598	0.173	0.607	0.183	0.618	0.193	0.630	0.206
0.49	0.608	0.184	0.618	0.194	0.630	0.205	0.643	0.218
0.50	0.619	0.194	0.629	0.205	0.641	0.217	0.655	0.231



Εφαρμογή



Εφαρμογή



Εφαρμογή

The screenshot shows a software interface for calculating the required reinforcement for a beam. The window title is "Υπολογισμός απαιτούμενου οπλισμού κάμψης δοκού".

Diagram: A beam of width b and height h is shown with effective depth d . The required reinforcement areas are A_{s1} (top) and A_{s2} (bottom). Internal forces N_{sd} and M_{sd} are indicated.

Input Fields:

- Ροπή κάμψης M_{sd} (kNm): 100
- Αξονικό φορτίο N_{sd} (kN): -200
- Ποιότητα χάλυβα: [Dropdown]
- Ποιότητα σκυροδέματος: [Dropdown]
- Πλάτος b (cm): 20
- Ύψος h (cm): 70
- Επικάλυψη (cm): 3
- Διάμετρος ράβδων: [Dropdown]

Table of Required Reinforcement:

Symbol	Variable	Description
μ	$lblm$	Ελάχιστος οπλισμός ανοίγματος $A_{s1, min} = lblAs1min$
ω_1	$lblOmega1$	Μέγιστος οπλισμός ανοίγματος $A_{s1, max} = lblAs1max$
ω_2	$lblOmega2$	Εφεδκούμενος οπλισμός $A_{s1} = lblAs1$
d_2/d	$lbld2d$	Απαιτούμενος οπλισμός $A_{s1} = lblAsreq1$
ρ_{min}	$lblrmin$	Βιβλόμενος οπλισμός $A_{s2} = lblAs2$
ρ_{max}	$lblrmax$	Απαιτούμενος οπλισμός $A_{s2} = lblAsreq2$

Buttons: Υπολογισμός (Calc), Άκυρο (Cancel)

Labels: frmDeamDesign, pic1, frame1, cmbCalc, cmbCancel



Εφαρμογή

The screenshot displays the Microsoft Visual Studio environment with a Windows Forms application titled "Υπολογισμός απαιτούμενου σπλισμού κάμψης δοκού". The application window contains a diagram of a beam with dimensions b , d , and h , and reinforcement areas A_{s1} and A_{s2} . The diagram shows a beam with a dashed horizontal line representing the neutral axis. A red arrow labeled N_{sd} points to the right, and a red curved arrow labeled M_{sd} indicates a clockwise moment. The beam is shown in two views: a side view on the left and a top view on the right. The top view shows the width b and the effective depth d . The reinforcement areas A_{s1} and A_{s2} are indicated by arrows pointing to the top and bottom reinforcement bars, respectively.

Below the diagram, there is a section titled "Απαιτούμενος σπλισμός" with a grid of labels: Label9, Label10, Label11, Label12, Label13, Label14, Label15, Label16, Label17, Label18, Label19, and Label20.

On the right side of the application window, there are input fields for the following parameters:

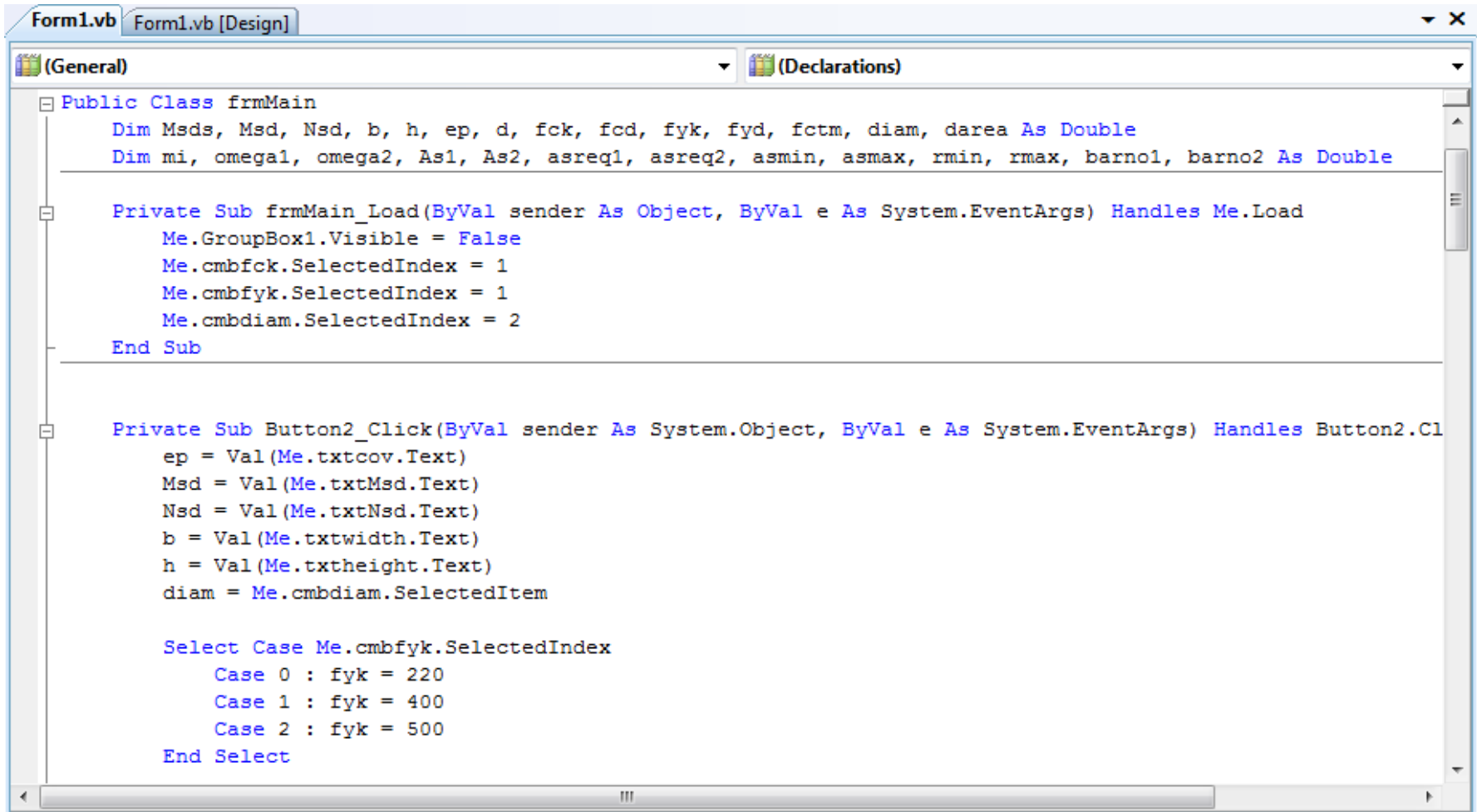
- Ροπή κάμψης $M_{sd}(kNm)$: 200
- Αξονικό φορτίο $N_{sd}(kN)$: 0
- Ποιότητα χάλυβα: [Dropdown menu]
- Ποιότητα σκυροδέματος: [Dropdown menu]
- Πλάτος $b(cm)$: 25
- Ύψος $h(cm)$: 60
- Επικάλυψη (cm) : 5
- Διάμετρος ράβδων: [Dropdown menu]

At the bottom of the application window, there are two buttons: "Υπολογισμός" and "Έξοδος".

The Visual Studio interface includes a Toolbox on the left with various Windows Forms controls, and an Error List at the bottom showing 0 Errors, 0 Warnings, and 0 Messages.



Εφαρμογή



```
Public Class frmMain
    Dim Msds, Msd, Nsd, b, h, ep, d, fck, fcd, fyk, fyd, fctm, diam, darea As Double
    Dim mi, omega1, omega2, As1, As2, asreq1, asreq2, asmin, asmax, rmin, rmax, barno1, barno2 As Double

    Private Sub frmMain_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Load
        Me.GroupBox1.Visible = False
        Me.cmbfck.SelectedIndex = 1
        Me.cmbfyk.SelectedIndex = 1
        Me.cmbdiam.SelectedIndex = 2
    End Sub

    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button2.Cl
        ep = Val(Me.txtcov.Text)
        Msd = Val(Me.txtMsd.Text)
        Nsd = Val(Me.txtNsd.Text)
        b = Val(Me.txtwidth.Text)
        h = Val(Me.txtheight.Text)
        diam = Me.cmbdiam.SelectedItem

        Select Case Me.cmbfyk.SelectedIndex
            Case 0 : fyk = 220
            Case 1 : fyk = 400
            Case 2 : fyk = 500
        End Select
    End Sub
End Class
```

Εφαρμογή

```
Form1.vb Form1.vb [Design]
(General) (Declarations)

Select Case Me.cmbfck.SelectedIndex
    Case 0 : fck = 12 : fctm = 1.6
    Case 1 : fck = 16 : fctm = 1.9
    Case 2 : fck = 20 : fctm = 2.2
    Case 3 : fck = 25 : fctm = 2.6
    Case 4 : fck = 30 : fctm = 2.9
End Select

darea = Math.Round(3.14 * (diam / 10) ^ 2 / 4, 2)

fyd = fyk / 1.15
fcd = fck / 1.5
d = h - ep
Msds = Msd - Nsd * (h / 2 - ep) / 100
mi = Math.Round(Msds * 1000 / (b * d * d * fcd), 5)

Select Case fyk
    Case 220 : Select Case ep / d
        Case 0 To 0.075 : omega1 = 1.0513 * mi + 0.0977 : omega2 = 1.0541 * mi - 0.3377
        Case 0.075 To 0.125 : omega1 = 1.1121 * mi + 0.0882 : omega2 = 1.1112 * mi - 0.3664
        Case 0.125 To 0.175 : omega1 = 1.1939 * mi + 0.0691 : omega2 = 1.1592 * mi - 0.3908
        Case 0.175 To 0.225 : omega1 = 1.2986 * mi + 0.0405 : omega2 = 1.1979 * mi - 4109
    End Select
    Case 400 : Select Case ep / d
```



Εφαρμογή

```
Form1.vb Form1.vb [Design]
(General) (Declarations)
Case 400 : Select Case ep / d
    Case 0 To 0.075 : omega1 = 1.0513 * mi + 0.0977 : omega2 = 1.0541 * mi - 0.3377
    Case 0.075 To 0.125 : omega1 = 1.1121 * mi + 0.0882 : omega2 = 1.1112 * mi - 0.3664
    Case 0.125 To 0.175 : omega1 = 1.1809 * mi + 0.0672 : omega2 = 1.1717 * mi - 0.3884
    Case 0.175 To 0.225 : omega1 = 1.2551 * mi + 0.0426 : omega2 = 1.2476 * mi - 0.4138
End Select
Case 500 : Select Case ep / d
    Case 0 To 0.075 : omega1 = 1.0537 * mi + 0.0919 : omega2 = 1.0532 * mi - 0.3324
    Case 0.075 To 0.125 : omega1 = 1.1117 * mi + 0.0735 : omega2 = 1.1125 * mi - 0.3511
    Case 0.125 To 0.175 : omega1 = 1.1765 * mi + 0.0532 : omega2 = 1.1748 * mi - 0.3706
    Case 0.175 To 0.225 : omega1 = 1.2495 * mi + 0.0304 : omega2 = 1.2505 * mi - 0.3944
End Select
End Select

If mi >= 0.33 Then
    As1 = omega1 * b * d * fcd / fyd + Nsd * 10 / fyd
    As2 = omega2 * b * d * fcd / fyd
Else
    If mi > 0.01 Then
        omega1 = 8.3708 * mi ^ 3 - 2.6563 * mi ^ 2 + 1.3936 * mi - 0.0104
        As1 = omega1 * b * d * fcd / fyd + Nsd / fyd
        omega2 = 0
        As2 = 0
    Else
        omega1 = mi
        As1 = omega1 * b * d * fcd / fyd + Nsd / fyd
    End If
End If
```



Εφαρμογή

```
Form1.vb Form1.vb [Design]
(General) (Declarations)

Else
    omega1 = mi
    As1 = omega1 * b * d * fcd / fyd + Nsd / fyd
    omega2 = 0
    As2 = 0
End If
End If

rmin = Math.Round(0.5 * fctm / fyd, 6)
rmax = 0.04
asmin = Math.Round(rmin * b * d, 2)
asmax = Math.Round(rmax * b * d, 2)

If As1 < asmin Then As1 = asmin
If As1 > asmax Then As1 = asmax

barno1 = Int(As1 / darea) + 1
barno2 = Int(As2 / darea) + 1

Me.GroupBox1.Visible = True
Me.lblm.Text = "μ = " & mi
Me.lblomega1.Text = "ω1 = " & Math.Round(omega1, 4)
Me.lblomega2.Text = "ω2 = " & Math.Round(omega2, 4)
Me.lbd2d.Text = "d2/d = " & Math.Round(ep / d, 2)
Me.lblrmin.Text = "ρmin = " & Math.Round(rmin * 100, 2) & "%"
Me.lblrmax.Text = "ρmax = " & Math.Round(rmax * 100, 2) & "%"
```


Εφαρμογή

```
Form1.vb Form1.vb [Design]
(General) (Declarations)
barno2 = Int(As2 / darea) + 1

Me.GroupBox1.Visible = True
Me.lblm.Text = "μ = " & mi
Me.lblomega1.Text = "ω1 = " & Math.Round(omega1, 4)
Me.lblomega2.Text = "ω2 = " & Math.Round(omega2, 4)
Me.lblld2d.Text = "d2/d = " & Math.Round(ep / d, 2)
Me.lblrmin.Text = "ρmin = " & Math.Round(rmin * 100, 2) & "%"
Me.lblrmax.Text = "ρmax = " & Math.Round(rmax * 100, 2) & "%"
Me.lblasmin.Text = "Ελάχιστος οπλισμός ανοίγματος As1,min = " & asmin & " cm2"
Me.lblasmax.Text = "Μέγιστος οπλισμός ανοίγματος As1,max = " & asmax & " cm2"
Me.lblas1.Text = "Εφελκυσόμενος οπλισμός As1 = " & Math.Round(As1, 2) & " cm2"
Me.lblas2.Text = "Θλιβόμενος οπλισμός As2 = " & Math.Round(As2, 2) & " cm2"
Me.lblas1req.Text = barno1 & "φ" & diam & " = (" & Math.Round(barno1 * darea, 2) & " cm2)"
Me.lblas2req.Text = barno2 & "φ" & diam & " = (" & Math.Round(barno2 * darea, 2) & " cm2)"

If As2 = 0 Then
    Me.lblas2.Visible = False
    Me.lblas2req.Visible = False
Else
    Me.lblas2.Visible = True
    Me.lblas2req.Visible = True
End If

End Sub
```



Σημείωμα Αναφοράς

Copyright Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης, Αναστάσιος Σέξτος
«Τεχνικές Προγραμματισμού και Χρήση Λογισμικού Η/Υ στις Κατασκευές.
Διαδικασίες λογικών αποφάσεων και βρόγχων εργασιών». Έκδοση: 1.0.
Θεσσαλονίκη 2014. Διαθέσιμο από τη δικτυακή διεύθυνση:

<http://eclass.auth.gr/courses/OCRS322/>



Σημείωμα Αδειοδότησης

Το παρόν υλικό διατίθεται με τους όρους της άδειας χρήσης Creative Commons Αναφορά - Παρόμοια Διανομή [1] ή μεταγενέστερη, Διεθνής Έκδοση. Εξαιρούνται τα αυτοτελή έργα τρίτων π.χ. φωτογραφίες, διαγράμματα κ.λ.π., τα οποία εμπεριέχονται σε αυτό και τα οποία αναφέρονται μαζί με τους όρους χρήσης τους στο «Σημείωμα Χρήσης Έργων Τρίτων».



Ο δικαιούχος μπορεί να παρέχει στον αδειοδόχο ξεχωριστή άδεια να χρησιμοποιεί το έργο για εμπορική χρήση, εφόσον αυτό του ζητηθεί.

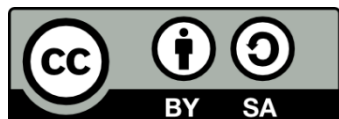
[1] <http://creativecommons.org/licenses/by-sa/4.0/>





Τέλος Ενότητας

Επεξεργασία: Ολυμπία Τασκάρη
Θεσσαλονίκη, <Ημερομηνία>



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ



ΑΡΙΣΤΟΤΕΛΕΙΟ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΘΕΣΣΑΛΟΝΙΚΗΣ

Σημειώματα

Διατήρηση Σημειωμάτων

Οποιαδήποτε αναπαραγωγή ή διασκευή του υλικού θα πρέπει να συμπεριλαμβάνει:

- το Σημείωμα Αναφοράς
- το Σημείωμα Αδειοδότησης
- τη δήλωση Διατήρησης Σημειωμάτων
- το Σημείωμα Χρήσης Έργων Τρίτων (εφόσον υπάρχει)

μαζί με τους συνοδευόμενους υπερσυνδέσμους.

